**3GPP TSG-SA WG6 Meeting #46-e S6-212600Rev1**

**e-meeting, 15th – 23rd November 2021 (revision of S6-21xxxx)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.1* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  |  | **CR** |  | **rev** |  | **Current version:** | **17.8.0** |  |
|  | | | | | | | | |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* | | | | | | | | |
|  | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME | **X** | Radio Access Network |  | Core Network | **X** |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | Clarification on the use of MCData notification server(s) | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | at&t | | | | | | | | | |
| ***Source to TSG:*** | S6 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | eMCData3 | | | | |  | ***Date:*** | | | 11-02-2021 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | F |  | | | | | ***Release:*** | | | Rel-17 |
|  | *Use one of the following categories:* ***F*** *(correction)* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | | | | | | | *Use one of the following releases: Rel-8 (Release 8) Rel-9 (Release 9) Rel-10 (Release 10) Rel-11 (Release 11) … Rel-15 (Release 15) Rel-16 (Release 16) Rel-17 (Release 17) Rel-18 (Release 18)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | Whether to support multiple MCData notification servers and how they are used were discussed by CT1 in their stage 3 development. This CR is intended to clarify if multiple MCData notification servers will be supported and how they should be used. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | 1. Add clarification to the MCData notification server in clause 6.4.3.5. 2. Add clarification to the Message notification client in clause 6.4.3.6. 3. Correct typos in figure 7.13.3.17.3-1, MCData notification client should be Message notification client. 4. Change the MCData notification server parameter in table A.5-2 to support multiple in deployment. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | There will be no guidance to related stage 3 development. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 6.4.3.5, 6.4.3.6, 7.13.3.17.3, A.5 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  | **X** | Other core specifications | | | | TS/TR ... CR ... | | |
| ***affected:*** | |  | **X** | Test specifications | | | | TS/TR ... CR ... | | |
| ***(show related CRs)*** | |  | **X** | O&M Specifications | | | | TS/TR ... CR ... | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

\* \* \* First Change \* \* \*

#### 6.4.3.5 MCData notification server

The MCData notification server provides the centralized notification function in the network. The MCData notification server allows an application (e.g. resident in the UE) to create a communication channel to receive real-time notifications from the network in either Pull or Push mode. Depending on the channel type created, the MCData notification server provides the application a callback endpoint (i.e. URL) and may also provide a channel endpoint (i.e. URL). The application communicates the callback endpoint information to the application server (i.e. network enabler) for it to use in sending to the MCData notification server the events for delivery to the application. Depending on the type of channel created by the application the delivery of the notifications from the MCData notification server to the application may be via a Pull or Push method. If Pull method is used, then the application shall use the provided channel endpoint to pull the notifications from the MCData notification server. However, if a Push notification delivery is used, then the MCData notification server asynchronously delivers the events received from the application server to the application through a PUSH Enabler server. The MCData notification server provides a consistent way to deliver notifications by all services to reduce the complexity of service logic on the application server.

It depends on deployment, if multiple MCData notification servers aredeployed.

#### 6.4.3.6 Message notification client

The Message notification client is used to request the notification service from the MCData notification server. Once the notification service request is authorized by the MCData notification server, the Message notification client will communicate the callback endpoint, received from the MCData notification server, to the MCData message store to be used for notification message delivery.

When multiple MCData notification servers are deployed, the Message notification client shall select one for notification service at any given time. How the Message notification client selects the appropriate MCData notification server from multiple available MCData notification servers for service is implementation specific.

\* \* \* Next Change \* \* \*

##### 7.13.3.17.3 Procedure using MCData notification server

The procedure in figure 7.13.3.17.3-1 describes how the MCData message store notifies the message notification client, using a MCData notification server, that there are new objects in the MCData message store needing to be synchronized. This procedure uses a web base notification mechanism in wide deployment today. The Message notification client requests the notification service from the MCData notification server and the MCData notification server returns with two URLs; one used by the service client to inform the service server where to send notification messages and the other one to use by the service client to PULL notification messages from the MCData notification server.

Pre-conditions:

1. The MCData user has an account with the MCData message store.

2. A successful authentication and authorization have been performed between the message store client and the MCData message store.

3. The Message store client doesn't have an ongoing session with the MCData message store.

4. The trust relationship between the MCData notification server and the MCData message store has been established.

5. The MCData notification server has a trust relationship and connection with the PUSH Enabler server.



Figure 7.13.3.17.3-1: Notify client to synchronize through MCData notification server

1. The Message notification client wants to create notification channels (i.e. endpoint URLs) to be used by the MCData message store to send notification messages and sends a Create notification channel request to the MCData notification server. The desired validity duration for the channels to be used and the notification channel type (PUSH or PULL) are included in the request.

2. The MCData notification server authenticates the Message notification client and authorizes its request.

3. The MCData notification server sends the Message notification client the Create notification channel response with the endpoint URLs that will be used by the MCData message store to send the notification messages and the Message notification client to receive the notification messages. The MCData notification server also includes what is the valid duration for these endpoint URLs to be used in the response.

4. If the notification type is PULL method, the message notification client sends the Open notification channel to the MCData notification server to start receiving the notification message. For certain PUSH method notification type (such as WebSockets) the message notification client requests the MCData notification server to start the PUSH notification service with its specific protocol that is outside the scope of this specification.

5. The message store client sends the Subscribe for notification request to the MCData message store asking to be notified if there are changes to its message store account. The callback URL returned from the MCData notification server in step 3 is included in the request for the MCData message store to use to send notification messages.

6. The MCData message store sends the Subscribe for notification response to the message store client to acknowledge the request.

7. The MCData user's message store account has changed and the MCData message store generates a notification message.

8. Using the callback URL, the MCData message store sends the notification message to the MCData notification server.

9. If the delivery method is PULL, the MCData notification server sends the notification message to the message notification client over the opened notification channel. If the delivery method is PUSH, the MCData notification server sends the notification message to the PUSH Enabler server (not shown in the figure) to deliver to the message notification client.

NOTE: The PUSH Enabler server is implementation specific and outside the scope of this specification.

The procedure in figure 7.13.3.17.3-2 describes how the message notification client updates the validity duration of a notification channel and subscription to avoid its expiration, i.e. to extend its lifetime.

Pre-conditions:

1. A notification channel has already been requested and established between the message notification client and MCData notification server.

2. The message store client has a successful notification subscription with the MCData message store.

3. The validity duration of the notification channel is about to expire.



Figure 7.13.3.17.3-2: Update a notification channel

1. The message notification client sends the Update notification channel request, including the desired new validity duration, to the MCData notification server.

2. The MCData notification server grants the request and sends the Update notification channel response to the message notification client. The new validity duration is included in the response.

3. The message store client sends the Update notification subscription request to the MCData message store with the new validity duration received from the MCData notification server in step 2.

4. The MCData message store sends the Update notification subscription response to the message store client and confirms the new validity duration.

The procedure in figure 7.13.3.17.3-3 describes how the message notification client delete a notification channel and subscription that is no longer needed.

Pre-conditions:

1. A notification channel has already been requested and established between the message notification client and MCData notification server.

2. The message store client has a successful notification subscription with the MCData message store.

3. The MCData user no longer wants to receive notifications from the MCData message store.



Figure 7.13.3.17.3-3: Delete a notification channel

1. The message store client decides to stop receiving notifications from the MCData message store and sends the Delete notification subscription request to the MCData message store.

2. The MCData message store acknowledges the request and sends the Delete notification subscription response to the message store client.

3. The message notification client sends the Delete notification channel request to the MCData notification server.

4. The MCData notification server acknowledges the request and sends the Delete notification channel response to the message notification client.

\* \* \* Next Change \* \* \*

# A.5 MCData service configuration data

The general aspects of MC service configuration are specified in 3GPP TS 23.280 [5]. The MCData service configuration data is stored in the MCData server.

Tables A.5-1 and A.5-2 describe the configuration data required to support the use of on-network MCData service. Tables A.5-1 and A.5-3 describe the configuration data required to support the use of off-network MCData service. Data in tables A.5-1 and A.5-3 can be configured offline using the CSC-11 reference point.

Table A.5-1: MCData service configuration data (on and off network)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Reference | Parameter description | MCData UE | MCData Server | Configuration management server |

Table A.5-2: MCData service configuration data (on network)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Reference | Parameter description | MCData UE | MCData Server | Configuration management server |
| Subclause 6.2.2 of 3GPP TS 22.282 [3] | Transmission and reception control |  |  |  |
|  | > Maximum data size for SDS | Y | Y | Y |
|  | > Maximum payload data size for SDS over signalling control plane (see NOTE 1) | Y | Y | Y |
|  | > Maximum data size for FD | Y | Y | Y |
| [R-6.2.2.1-002d],  [R-6.2.2.4-003] of 3GPP TS 22.282 [3] | > Time limit for the temporarily stored data waiting to be delivered to a receiving user | N | Y | Y |
| [R-6.2.2.3-001] of 3GPP TS 22.282 [3] | > Timer for periodic announcement with the list of available recently invited data group communications | N | Y | Y |
|  | > Maximum data size for auto-receive | N | Y | Y |
|  | List of functional alias identities |  |  |  |
| [R-5.9a-005] of 3GPP TS 22.280 [17] | > Functional alias | N | Y | Y |
| [R-5.9a-005] of 3GPP TS 22.280 [17] | >> Limit number of simultaneous activations | N | Y | Y |
| [R-5.9a-005] of 3GPP TS 22.280 [17] | >> This functional alias can be taken over | N | Y | Y |
|  | >> List of users who can activate this functional alias |  |  |  |
| [R-5.9a-005] of 3GPP TS 22.280 [17] | >>> MCData ID | N | Y | Y |
| [R-5.9a-016] of 3GPP TS 22.280 [17] | >> Communication priority (see NOTE 2) | N | Y | Y |
| [R-5.10-001a] of 3GPP TS 22.280 [2] | Maximum number of successful simultaneous service authorizations of clients from a user | N | Y | Y |
|  | MCData notification server |  |  |  |
|  | > Server URI(s) | Y | Y | Y |
| NOTE 1: The maximum payload data size for SDS over signalling control plane shall be less than or equal to the maximum data size for SDS.  NOTE 2: The usage of this parameter by the MCData server is up to implementation. | | | | |

Table A.5-3: MCData service configuration data (off network)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Reference | Parameter description | MCData UE | MCData Server | Configuration management server |
| Subclause 10.10 of 3GPP TS 23.280 [5] | Default ProSe Per-Packet priority (as specified in 3GPP TS 23.303 [7]) values |  |  |  |
|  | > MCData one-to-one call signalling | Y | N | Y |
|  | > MCData one-to-one call media | Y | N | Y |